BEST AVAILABLE COPY



5/6

Import and decrypt Draw around the Choose the material an image of a tooth to be matched (ceramic) which will be used patient's mouth (selection) to create the prothesis Create another image (extraction) using the pixel values from the selection Create an internal palette from the ceramic table and apply the palette to the extraction using octree quantisation For each pixel in the extraction determine a range key using following algorithm: The range key value consists of three elements: red component value divided by 16 green component value divided by 16 3. blue component value divided by 16 (In the above the range key values are therefore:

0 for component values 0-15

1 for component values 16-31

2 for component values 32-47

3 for component values 48-63

and so on)

Swap the range key value with a vivid colour value which is readily identifiable by the eye, such as yellow, cyan, magenta etc.

Enter information about this pixel into an internal array of the following structure:

Original pixel Red Green Blue Range key Count of pixels |Component|Component|value colour in this range

(For each pixel: If, in the array, the range key value does not yet exist, a new element is created, however, if the range key value already exists, the count is increased by one)

In a duplicate image (contour) of the extraction, replace the corresponding pixel with the one of the vivid colour



to Fig. 5B



6/6

from Fig. 5A



Once all pixels in the extraction have been processed, build an external table in an SQL compliant database of the same structure as the internal array and populate it with the contents of the internal array

Using stored SQL procedures in the database determine the dominant (most frequently occuring) original pixel value in each range key and create an internal array (key array)

For each element in the key array draw a vivid colour rectangle as a key underneath the contour image and identify it with the shade relevant to that key colour

Fig. 5B